



Input and state estimation exploiting input sparsity

Sophie Fosson, Federica Garin, Sebin Gracy, Alain Kibangou, Dennis Swart

► To cite this version:

Sophie Fosson, Federica Garin, Sebin Gracy, Alain Kibangou, Dennis Swart. Input and state estimation exploiting input sparsity. ECC 2019 - 18th European Control Conference, Jun 2019, Naples, Italy. pp.2344-2349, 10.23919/ECC.2019.8795699 . hal-02094213

HAL Id: hal-02094213

<https://hal.inria.fr/hal-02094213>

Submitted on 9 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Input and state estimation exploiting input sparsity

Sophie M. Fosson¹, Federica Garin², Sebin Gracy², Alain Y. Kibangou², Dennis Swart^{3,2}

Abstract—Motivated by cyber-physical security applications, we face the problem of estimating the state and the input of a linear system, where the input may represent the presence of adversarial attacks. We consider the case where classical filters cannot be used, because the number of measurements is too low, for example it is lower than the size of the input vector. If the input, although of large size, is known to be sparse, the problem can be tackled using techniques from compressed sensing theory. In this paper, we propose a recursive estimator, based on compressed sensing and Kalman-like filtering, which is able to reconstruct both the state and the input from noisy, compressed measurements. The proposed algorithm is proved to be feasible and numerically efficient, and simulations show a good recovery accuracy with respect to an oracle estimator.

I. INTRODUCTION

The problem of jointly estimating the state and the input of a linear system has attracted a lot of attention in the last years, mainly motivated by cyber-physical security applications. If a malicious agent injects an input to the system, in order to disrupt its functioning, the system controller might need to keep estimating the system state (despite the presence of the input), as well as to estimate the unknown input, so as to be able to counteract it, and preserve the system behaviour. The relevance of this problem for cyber-physical systems has been highlighted, *e.g.*, in [1], [2], [3], [4], where the authors characterize the systems where zero-dynamics attacks occur (namely, attacks such that the input and state estimation is not possible). The classical solution to the problem of joint input and state estimation is a Kalman-like filter [5], [6].

Without any limitation on the input, *e.g.*, on its size, the problem would require output measurements from every single state. However, in many cyber-physical security applications, one can make assumptions about realistic constraints that the attacker might encounter [7]. In the recent literature, the attacks are often supposed to be *sparse*, *i.e.*, the attacker can inject an input only in few places [8], [9], [10], [11], [12], [13]. This assumption is particularly meaningful when the input injection requires physical tampering in the location of each actuator, or requires a cyber attack of independent local units, so that the attacker can only make one, or very few such actions. In linear systems, this can be modeled as an input vector of large size, as large as the number of vulnerable states, but sparse, *i.e.*, with only few non-zero

entries. The support, that is, the positions of the non-zero entries, is not known and has to be detected.

In [10], [11], [12], [13], the problem of sparse input is tackled in terms of state estimation and control of the system, while the estimation of the input is not done. However, it might be important to estimate the input as well, in order to detect and isolate the attack, *e.g.*, by switching off the attacked actuators.

With this motivation, the aim of this paper is to propose a method to estimate both the state and the input, under a sparsity assumption for the input. This scenario has been considered in [14], where the authors study the theoretical conditions for unique input reconstruction, and propose two algorithms: the first one being batch (waiting for the whole output sequence before recovering the state and input sequences), and the second one being recursive. In that work, however, noise is not taken into account, which is a severe limitation for practical purposes. More precisely, the algorithm proposed in [14] cannot be extended to the noisy case and it is not tolerant to noise, as shown also in our numerical experiments (Section IV). A noise-free setting is considered also in [8], [9], where the identification of attacks is studied for continuous-time linear systems.

Our contribution is to provide an algorithm for joint input and state estimation, which is able to deal with noise. Specifically, we propose a Kalman filtering approach; moreover, assuming the input sparsity, we aim at performing estimation from compressed measurements, which is possible thanks to compressed sensing (CS) [15], [16], [17]. This is a recent theory in signal processing which allows to achieve unique reconstruction of a sparse signal with a number of samples below the Nyquist rate, under some technical conditions on the measurements. Since its advent, CS has been mostly focused on static problems, where signal reconstruction is done without any knowledge of the signal dynamics. In the last years, however, an increasing attention to time-varying problems has arisen, motivated by the ubiquitous applications. In the signal processing community, some research has been devoted to dynamic CS, where the unknown sparse signal is assumed to be time-varying, [18], [19], [20], [21]. In the framework of linear dynamical systems, instead, a line of research where CS has been combined with Kalman-like estimation has emerged, for systems where the state is assumed to be sparse, see [22], [23], [24], [25] and the references therein.

Differently from these papers, we do not assume sparsity of the state, but rather sparsity of the input. In particular, we use CS to recover the sparse input from a number of measurements smaller than the number of vulnerable entries.

¹ DAUIN, Politecnico di Torino, Italy.
sophie.fosson@polito.it

² Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, GIPSA-Lab, Grenoble, France.

federica.garin@inria.fr, sebin.gracy@inria.fr,
alain.kibangou@univ-grenoble-alpes.fr

³ University of Twente, Enschede, The Netherlands.

d.f.h.swart@student.utwente.nl

As discussed later, this is not achievable with classical input and state estimation approaches [5].

The paper is organized as follows. In Section II, we introduce the mathematical model and the necessary elements of CS. In Section III, we propose our algorithm and we discuss its theoretical properties. In IV, we show some numerical results, that show the good performance obtained by the proposed method when compared to an oracle estimator. Finally, some conclusions are drawn.

II. PROBLEM STATEMENT AND PRELIMINARIES

The problem that we aim to tackle can be formulated as follows. We consider a linear system

$$\begin{cases} x_{k+1} = Ax_k + Bu_k + w_k \\ y_k = Cx_k + v_k \end{cases} \quad (1)$$

where $x_k \in \mathbb{R}^n$, $y_k \in \mathbb{R}^m$, and $u_k \in \mathbb{R}^p$ are the state, output, and input, respectively. The input u_k is assumed to be sparse, *i.e.*, it has only $s_k \ll p$ non-zero entries. Its support is not known and might change at each k . As discussed in the the previous section, this can model a cyber-attack which can affect a small number of vulnerable nodes at each time instant. The process noise w_k and the measurement noise v_k are supposed to be zero-mean, uncorrelated, gaussian, with covariance matrices Q and R , respectively. Concerning u_k , we have no knowledge about its dynamical evolution in time, nor about its statistical properties. Notice that u_k is an unknown external input, while the possible closed-loop control input is omitted here, since it is known by the system controller, and it can be easily subtracted out from the state update equation.

The goal is to develop a recursive (Kalman-like) algorithm, which is able, at time k and given y_k , to promptly output estimates \hat{x}_k and \hat{u}_{k-1} of x_k and u_{k-1} , so that an attack can be readily detected and countermeasures can be taken, while monitoring of the system state keeps properly running. This problem is known as delay-1 input and state estimation.

We remark that the sparsity assumption cannot be expressed simply imposing a small input size p , because this would imply assuming perfect knowledge of the locations of the attacks. Instead, we rather propose the following. We describe with B some knowledge about all the possible ways the attacker might locally influence the states. In particular, we consider B of the form $B = \begin{bmatrix} I_p \\ 0 \end{bmatrix}$, which describes the fact that p states are vulnerable to local attacks (an input attack might be injected in either of these states, individually), and $n - p$ states are reliable and cannot be attacked, *e.g.*, because they have a stronger protection, or because they are internal states and no actuator can directly act on them.

A. Classical joint input and state estimation

The problem of jointly estimating the state and the unknown input (with no extra assumption on the input sparsity) is classical. The conditions under which a stable estimator exists are discussed in [26] and [6], and are the following: the system is delay-1 left-invertible (namely CB has full

column rank), and is strongly detectable, namely $\begin{bmatrix} A - zI & B \\ C & 0 \end{bmatrix}$ has full column rank for all $z \in \mathbb{C}$ with $|z| \geq 1$.

In [5], an optimal (minimum-variance unbiased) estimator is proposed. This algorithm requires the delay-1 left-invertibility ($\text{rank}(CB) = p$). Notice that $\text{rank}(CB) = p$ implies $\text{rank}(C) = p$, and in particular $m \geq p$, which excludes the possibility of estimation from compressed measurements ($m < p$), thus the use of possible prior information on sparsity and CS. As discussed in [6], this filter is stable (*i.e.*, covariance remains bounded) if the system is strongly detectable.

B. Compressed sensing in a nutshell

The problem posed by CS is to find the sparsest solution (*i.e.*, the solution with maximum number of null components) to the underdetermined linear system of equations $z = \Phi\alpha$, $\alpha \in \mathbb{R}^p$, $\Phi \in \mathbb{R}^{m,p}$, $m < p$, [15], [16]. The uniqueness of the solution α is guaranteed if the so-called sensing matrix Φ satisfies certain conditions, in terms of coherence or restricted isometry property (RIP) [17]. Since these conditions are difficult to prove for deterministic matrices, random matrices are generally considered. In particular, gaussian matrices are known to satisfy the RIP (see [17] for a deeper overview).

The CS problem is NP-hard, which makes the recovery a tough point. For this reason, convex formulations of CS have been studied [27], which are computationally affordable, *e.g.*, Basis Pursuit, for the noise-free case, and Basis Pursuit Denoising and Lasso, for the noisy case [17]. In particular, Lasso [28] is a least squares problem with an ℓ_1 -norm regularizer, which is known to promote sparsity and has long been known in statistical and machine learning communities, also for overdetermined problems.

C. Exploiting compressed sensing for input estimation

In [14], the idea of exploiting CS for joint reconstruction of state and sparse input is introduced. More precisely, CS is used in the input recovery phase to estimate the sparse input from compressed measurements ($m < p$). First, a batch algorithm is proposed, where the whole output sequence is collected and used for recovery; sufficient conditions for successful reconstruction are discussed. Then, an iterative algorithm is developed, which can be summarized as follows: given an initial state estimate \hat{x}_0 , at each time instant k , an estimate \hat{u}_{k-1} of the input u_{k-1} is obtained, using CS techniques, from the (underdetermined) linear system:

$$\tilde{y}_k = CBu$$

where $\tilde{y}_k = y_k - CA\hat{x}_{k-1}$. Then, the state is estimated as follows:

$$\hat{x}_k = A\hat{x}_{k-1} + B\hat{u}_{k-1}.$$

More precisely, assuming the absence of noise, \hat{u}_{k-1} is computed as the solution of the Basis Pursuit problem [17], which reads as follows: $\min_{u \in \mathbb{R}^p} \|u\|_1$ s.t. $\tilde{y}_k = CBu$.

In the noise-free case, if CB satisfies the properties for CS recovery, this algorithm provides the exact sequence of states and inputs. However, when noise is introduced, this

procedure is not stable: the state reconstruction is just the input-corrected prediction, while the output measurements are only used to estimate the input. No Kalman-like correction of the open-loop state prediction is performed, therefore, unless A is stable, errors are propagated (see Figure 4).

For this reason, the approach proposed in [14] is not feasible in case of noise, and no straightforward extension can be obtained from it to tolerate noise. In the next section, we introduce our novel approach to this problem.

III. PROPOSED ALGORITHM

Our goal is to provide a recursive algorithm, which is able

- to exploit the input sparsity and CS paradigm so as to deal with cases where CB does not have full column rank, as in [14] and differently from [5];
- to estimate state and input from compressed, noisy measurements, as in [5] and differently from [14].

In this section, we introduce the proposed algorithm and we illustrate its properties. In the rest of the paper, given a set $S \subseteq \{1, \dots, p\}$, with cardinality $|S| = s$, we denote by B_S the $n \times s$ matrix obtained selecting the s columns of B whose index belongs to S .

The proposed algorithm exploits a Kalman filtering approach as in [5], but with a peculiar feature: at each k , the support of the sparse input is estimated and only the columns of B associated with the estimated support are used. The complete procedure can be summarized as follows.

Algorithm - SISE (Sparse-input and state estimator)

Initialization: $\hat{x}_0, P_0 > 0$.

For $k \geq 1$, **read measurement** y_k **and compute:**

Preliminary calculations

$$\tilde{P}_k = AP_{k-1}A^T + Q \quad (2)$$

$$\tilde{R}_k = C\tilde{P}_kC^T + R \quad (3)$$

Innovation

$$\tilde{y}_k = y_k - CA\hat{x}_k \quad (4)$$

Input support estimate

\hat{S}_{k-1} = support of the CS solution u of:

$$\tilde{y}_k = CBu + e_k \quad (5)$$

where e_k is a noise with 0 mean and covariance \tilde{R}_k

$$B_{k-1} = B_{\hat{S}_{k-1}} \quad (6)$$

Input estimate

$$M_k = (B_{k-1}^T C^T \tilde{R}_k^{-1} C B_{k-1})^{-1} B_{k-1}^T C^T \tilde{R}_k^{-1} \quad (7)$$

$$\hat{u}_{k-1} = M_k \tilde{y}_k \quad (8)$$

State estimate: input-corrected prediction

$$\hat{x}_k^* = A\hat{x}_k + B_{k-1}\hat{u}_{k-1} \quad (9)$$

$$P_k^* = (I - B_{k-1}M_kC)\tilde{P}_k(I - B_{k-1}M_kC)^T + B_{k-1}M_kRM_k^T B_{k-1}^T \quad (10)$$

State estimate: update

$$K_k = \tilde{P}_k C^T \tilde{R}_k^{-1} \quad (11)$$

$$\hat{x}_k = \hat{x}_k^* + K_k(y_k - C\hat{x}_k^*) \quad (12)$$

$$P_k = P_k^* - K_k(P_k^* C^T - B_{k-1}M_kR)^T \quad (13)$$

The specific method to find the solution of (5), then its support, is discussed later in Section III-A.

The presence of the stochastic error e_k in (5) is explained as follows (see [5] for the complete proof). Given a state estimate \hat{x}_{k-1} with error $\tilde{x}_{k-1} := x_{k-1} - \hat{x}_{k-1}$ having zero mean and covariance P_{k-1} , the system equations (1) imply that

$$\tilde{y}_k = CBu_{k-1} + e_k, \quad (14)$$

where $\tilde{y}_k = y_k - CA\hat{x}_{k-1}$ is the innovation, and e_k is a noise, given by

$$e_k = C(A\tilde{x}_{k-1} + w_{k-1}) + v_k, \quad (15)$$

having zero mean and having a covariance \tilde{R}_k that can be computed from P_{k-1} using (2)–(3).

Remark 1: The algorithm proposed in [5] is analogous to SISE with no support detection: the whole matrix B is used in [5] instead of the submatrix B_{k-1} . The algorithm proposed in [5] produces \hat{u}_{k-1} by finding the best linear unbiased estimate of the solution of the noisy equation (14); clearly, this can be done only under the assumption that CB has full column rank. Analyzing the steps of the algorithm SISE, it can be noticed that this requirement is relaxed to require that CB_{k-1} has full column rank, which is clearly a weaker restriction.

This point is fundamental as it allows us to work with compressed measurements ($m < p$), and exploit CS techniques. When $m < p$, indeed, CB never has full column rank, while CB_{k-1} can have full column rank.

A. Input support recovery

In this paper, we propose to perform the step (5) of SISE using the Lasso formulation of CS, which reads as follows:

$$\min_{u \in \mathbb{R}^p} \|\tilde{y}_k - CBu\|_2^2 + \lambda \|u\|_1, \quad \lambda > 0.$$

The parameter λ balances the weight between the least squares term and the sparsity promoting regularizer: a larger λ is required in the presence of a larger noise, see [17, Chapter 3] and [29] for details.

If CB has full row rank, the solution of Lasso has at most m non-zero entries [30], and hence the number of columns of CB_{k-1} does not exceed the number of rows. This ensures that CB_{k-1} has full column rank for all k if $B = \begin{bmatrix} I_p \\ 0 \end{bmatrix}$ and C has random independent entries, namely when measurements are few linear combinations of all vulnerable states according to some randomly designed coefficients.

We remark that other CS recovery strategies could be tested. For example, the Orthogonal Matching Pursuit [31] is a greedy method which iteratively builds the estimate by introducing at each step a new element in the support. This method requires only s iterative steps to build a support of

dimension s . However, this method is known to require a larger number of measurements to be successful with respect to Lasso, and also it requires to know s . For this reason, we prefer to use Lasso for our experiments.

Moreover, in order to solve Lasso, we can use a very efficient iterative algorithm, the Alternating Direction Method of Multipliers (ADMM) [32] (see Section IV), which is more efficient than other convex optimization methods, such as interior points. ADMM rapidly converges to the minimum of Lasso functional, and it is very popular also for its natural predisposition to decentralization [33] and parallelization [34], which makes it efficient even for large-scale problems. These properties of ADMM perfectly match with our procedure, in which the support has to be evaluated at each k , therefore the rapidity of solution is definitely important.

We finally remark that in (5) the goal is to recover the support, and not the values of the non-zero entries. Therefore, solving a whole Lasso is somewhat redundant. For example, one might study strategies to stop the ADMM procedure once the support has been detected, which might save some time. Moreover, online methods have been recently proposed which allow to track time-varying sparse signals without solving a complete Lasso at each k [21]. This further accelerates the algorithm and opens the way for solving large-scale problems. These numerical improvements will be investigated in future work.

B. Support recovery guarantees

If the support recovery \hat{S}_k in step (5) is exact, for all k , then SISE is equivalent to the algorithm in [5] on a system with a known, time-varying B_k , and inherits its optimal properties: it gives minimum-variance unbiased estimates and it is a stable filter, provided that the corresponding time-varying system is uniformly strongly detectable [6, Theorem 5].

For this reason, it is fundamental to accurately identify the support. Using Lasso, guarantees of exact support recovery are given in presence of bounded or gaussian noise, see, e.g., [35], [29]. In our setting, the noise e_k in general is not bounded nor gaussian; it is gaussian in case the input support has been perfectly recovered in all previous time steps. In practice, after an initial transient, the noise magnitude is sufficiently small, which allows a successful support recovery most of the times (see Section IV).

We finally remark the following limitation of our algorithm: it requires that CB_k has full column rank, for all k . In the case where $B = \begin{bmatrix} I_p \\ 0 \end{bmatrix}$ and B_k has one column, the B_k 's columns form the canonical basis of \mathbb{R}^p . Therefore, the condition $\text{rank}(CB_k) = 1$ for all k amounts at asking that the first p columns of C are non-zero, which means that all the vulnerable states are taken into account into some measurement. This requirement might be restrictive for some specific applications, for example, in case of distant attacks [9, Example 1]. Differently from [9], our limitation is not a consequence of the ℓ_1 regularization, and remains true if the input reconstruction in (5) is obtained by exactly solving the ℓ_0 problem or even with an exact oracle. Indeed, the rank

condition on CB_k is necessary for the input estimation in (7)-(8). To overcome this limitation, one needs to consider different filters, where the input is reconstructed with a delay $\ell > 1$ [26], namely after measuring y_k one reconstructs $u_{k-\ell}$ instead of u_{k-1} as in SISE.

IV. SIMULATION RESULTS

In this section, we show the results of some simulations. The considered scenario is the following. A has independent, gaussian components $\sim N(0, 1/\sqrt{n})$; $B = \begin{bmatrix} I_p \\ 0 \end{bmatrix}$; C has independent, gaussian components $\sim N(0, 1)$. From a CS viewpoint, the sensing matrix CB is then gaussian, which guarantees good CS properties, [17], [29]. We fix the input sparsity level to $s = 1$, and we assume to know s . The problem is then to find the positions of the non-zero value. As an initial example, the support is assumed to be constant in time, although this information is not exploited in the algorithm. The initial condition is assumed to be a standard gaussian random variable $\sim N(0, 1)$. We finally fix noise covariances $R = Q = 0.01I$, and the non-zero entry of the input is such that the input-to-noise-ratio is 20dB.

In the first three experiments, we run the proposed SISE in a system with $m < p$, and we compare it to an oracle version of [5] which knows the exact support of the input. This works as a benchmark to evaluate the performance of SISE. In the fourth experiment, we show instead a comparison to [14]. The considered performance metrics for input and state are the relative errors $\|u_k - \hat{u}_k\|_2 / \|u_k\|_2$ and $\|x_k - \hat{x}_k\|_2 / \|x_k\|_2$. For each experiment, we show averaged results over 100 runs.

As mentioned before, different algorithms can be used to solve the CS problem; for these experiments, we use the Lasso formulation. The Lasso parameter λ is fixed to $\lambda = 1$. This choice follows the prescription in [29] for the design of λ based on the noise variance. In this problem, the noise is larger in the first steps, which would suggest a large λ . However, sometimes the noise is too large and would push a λ that cuts to zero all the input; therefore, we set it based on the value of the noise after the transient. In order to cope with the transient larger noise, in the first steps one might think to take more measurements, which would definitely reduce the effect of noise. This will be tested in future extended work.

To solve the Lasso, we implement ADMM [32], which is sufficiently fast in our setting. Running the experiments on MATLAB, on a PC with quad-core CPU @ 1.80GHz, RAM 16Gb, the time consumed for an iteration k is around 10^{-3} seconds. Given the Lasso solution, the support is estimated by picking the component with largest magnitude, as we assume to know that $s = 1$.

We show results for different dimensions. In the experiment 1 (Figure 1), we consider the case $p = 40$, $m = 20$, $n = 50$. We can see that, after some iterations, the support is correctly estimated in almost all the runs. The mean relative errors are of order 10^{-2} , very close to the benchmark. The 100% of correct support is achieved increasing the number of measurements to $m = 30$ (see experiment 2 in Figure 2). In this case, we are then able to provide an optimal estimation

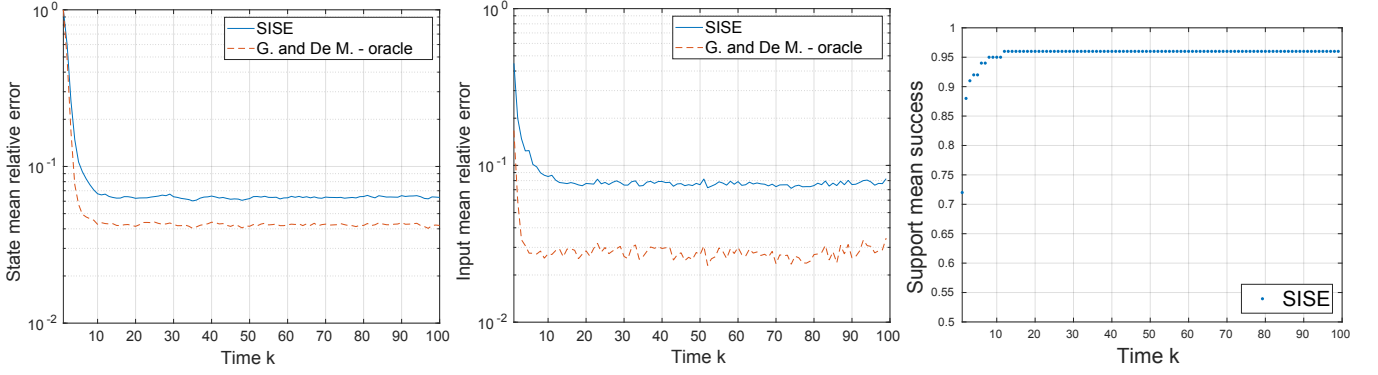


Fig. 1: Experiment 1: $p = 40$, $m = 20$, $n = 50$. The benchmark is an oracle version of [5] which knows the correct support.

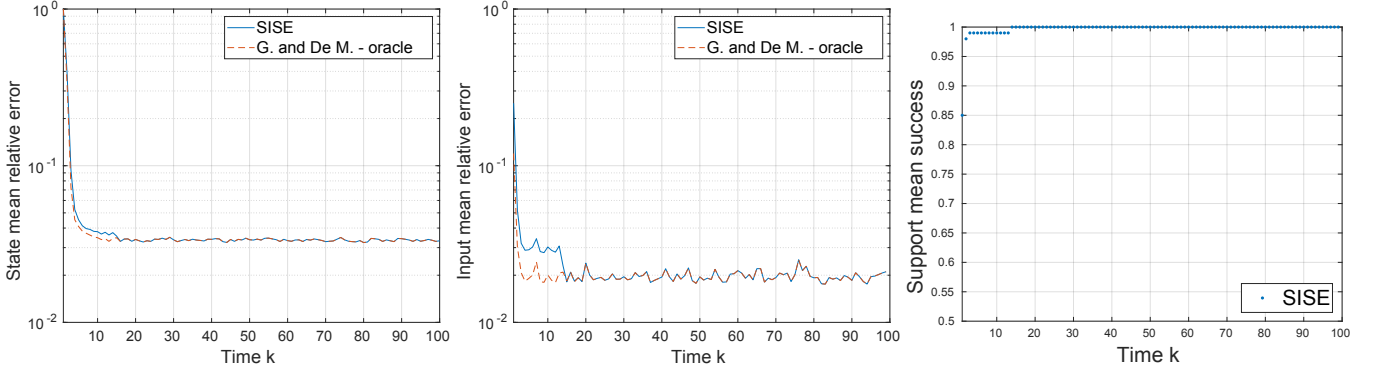


Fig. 2: Experiment 2: $p = 40$, $m = 30$, $n = 50$. The benchmark is an oracle version of [5] which knows the correct support. With $m = 30$, SISE achieves a perfect support estimation.

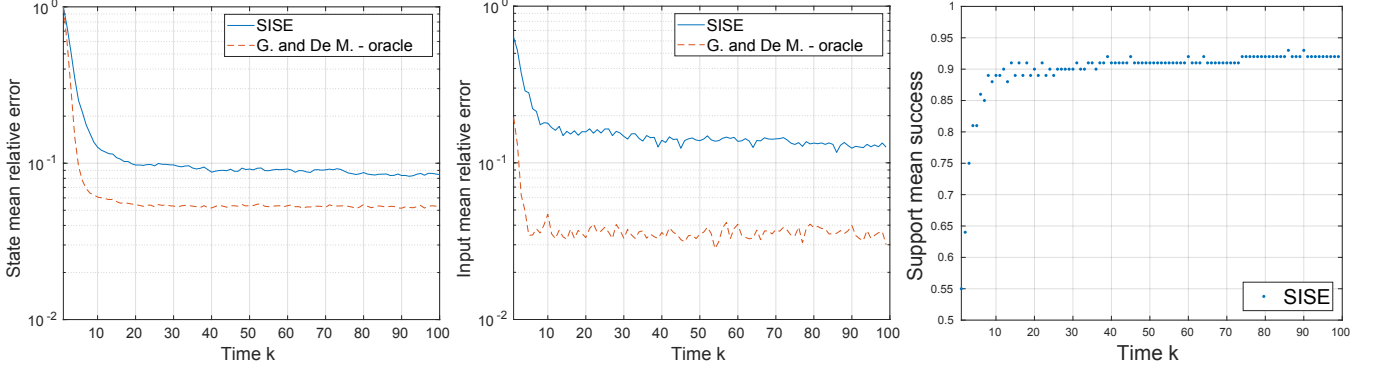


Fig. 3: Experiment 3: $p = 40$, $m = 20$, $n = 70$. The benchmark is an oracle version of [5] which knows the correct support. As expected, the estimation accuracy decreases with a larger n/m .

from compressed measurements, which was not possible with classical approach [5]. In experiment 3, we show that the estimation accuracy decreases when the number of states is large with respect to the number of measurements: in Figure 3 the case $p = 40$, $m = 20$, $n = 70$ is depicted.

Finally, experiment 4 in Figure 4 shows a comparison to [14], which is not robust to noise. In particular, we can appreciate that the estimation accuracy degrades in time, and the relative errors tend to grow unbounded. This shows that the approach of [14] is definitely not tolerant to noise.

V. CONCLUSIONS

In this paper, we have proposed a novel filter for sparse-input and state estimation in linear systems, in the presence of compressed measurements and noise. Our method leverages on compressed sensing techniques to estimate the input support, which provides some conditions to guarantee the success. Future work will include: correcting the covariance computation depending on the quality of the support reconstruction; using compressed sensing to estimate the input itself instead of the support only (with the difficulty to approximate the covariance of this non-linear estimate).

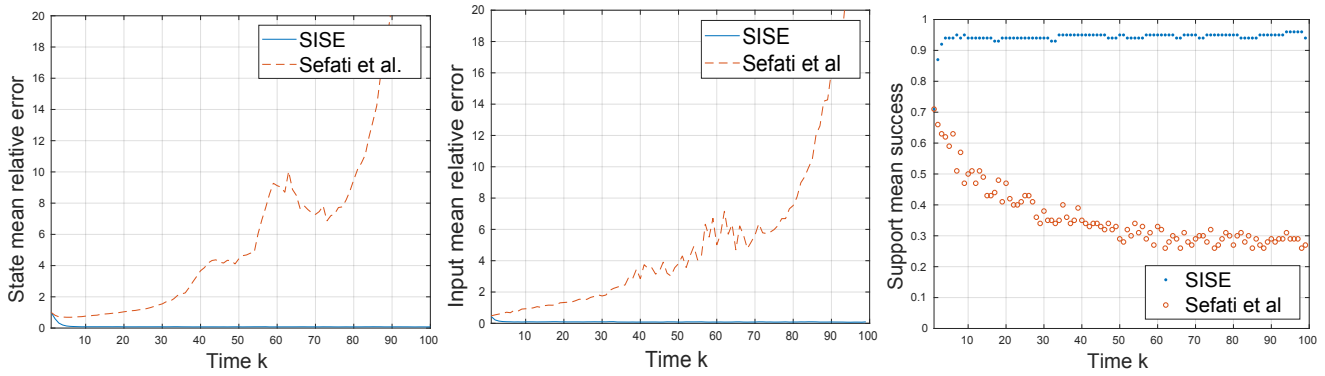


Fig. 4: Experiment 4: $p = 40$, $m = 20$, $n = 50$. We compare the proposed SISE to [14], which, as expected, is not robust to noise: its estimation capacity degrades with time.

REFERENCES

- [1] F. Pasqualetti, F. Dörfler, and F. Bullo, "Control-theoretic methods for cyberphysical security: Geometric principles for optimal cross-layer resilient control systems," *IEEE Control Syst.*, vol. 35, no. 1, pp. 110–127, 2015.
- [2] S. Weerakkody, X. Liu, and B. Sinopoli, "Robust structural analysis and design of distributed control systems to prevent zero dynamics attacks," in *Proc. IEEE Conf. Decision Control (CDC)*, 2017, pp. 1356–1361.
- [3] S. Weerakkody, X. Liu, S. H. Son, and B. Sinopoli, "A graph-theoretic characterization of perfect attackability for secure design of distributed control systems," *IEEE Trans. Control Netw. Syst.*, vol. 4, no. 1, pp. 60–70, 2017.
- [4] S. Gracy, F. Garin, and A. Y. Kibangou, "Structural and strongly structural input and state observability of linear network systems," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 4, pp. 2062–2072, 2018.
- [5] S. Gillijns and B. De Moor, "Unbiased minimum-variance input and state estimation for linear discrete-time systems," *Automatica*, vol. 43, no. 1, pp. 111–116, 2007.
- [6] S. Z. Yong, M. Zhu, and E. Frazzoli, "A unified filter for simultaneous input and state estimation of linear discrete-time stochastic systems," *Automatica*, vol. 63, pp. 321–329, 2016.
- [7] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "A secure control framework for resource-limited adversaries," *Automatica*, vol. 51, pp. 135–148, 2015.
- [8] F. Pasqualetti, F. Dörfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE Trans. Autom. Control*, vol. 58, no. 11, pp. 2715–2729, 2013.
- [9] —, "A divide-and-conquer approach to distributed attack identification," in *Proc. IEEE Conf. Decision Control*, 2015, pp. 5801–5807.
- [10] H. Fawzi, P. Tabuada, and S. Diggavi, "Secure estimation and control for cyber-physical systems under adversarial attacks," *IEEE Trans. Autom. Control*, vol. 59, no. 6, pp. 1454–1467, 2014.
- [11] Y. Shoukry and P. Tabuada, "Event-triggered state observers for sparse sensor noise/attacks," *IEEE Trans. Autom. Control*, vol. 61, no. 8, pp. 2079–2091, 2016.
- [12] C. S. Liu, J. Wu, C. N. Long, and Y. B. Wang, "Dynamic state recovery for cyber-physical systems under switching location attacks," *IEEE Trans. Control Netw. Syst.*, vol. 4, no. 1, pp. 14–22, 2017.
- [13] L. An and G. Yang, "Secure state estimation against sparse sensor attacks with adaptive switching mechanism," *IEEE Trans. Autom. Control*, vol. 63, no. 8, pp. 2596–2603, 2018.
- [14] S. Sefati, N. J. Cowan, and R. Vidal, "Linear systems with sparse inputs: Observability and input recovery," in *Proc. American Control Conf. (ACC)*, 2015, pp. 5251–5257.
- [15] D. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, pp. 1289–1306, 2006.
- [16] E. Candès and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, pp. 21–30, 2008.
- [17] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*. New York: Springer, 2013.
- [18] J. Ziniel and P. Schniter, "Dynamic compressive sensing of time-varying signals via approximate message passing," *IEEE Trans. Signal Process.*, vol. 61, no. 21, pp. 5270–5284, 2013.
- [19] A. Balavoine, C. J. Rozell, and J. Romberg, "Discrete and continuous-time soft-thresholding for dynamic signal recovery," *IEEE Trans. Signal Process.*, vol. 63, no. 12, pp. 3165–3176, 2015.
- [20] A. S. Charles, A. Balavoine, and C. J. Rozell, "Dynamic filtering of time-varying sparse signals via ℓ_1 minimization," *IEEE Trans. Signal Process.*, vol. 64, no. 21, pp. 5644–5656, 2016.
- [21] S. M. Fosson, "Online optimization in dynamic environments: a regret analysis for sparse problems," in *Proc. IEEE Conf. Decision Control (CDC)*, 2018, pp. 7225–7230.
- [22] N. Vaswani, "Kalman filtered compressed sensing," in *IEEE Int. Conf. on Image Processing (ICIP)*, 2008, pp. 893–896.
- [23] A. Carmi, P. Gurfil, and D. Kanevsky, "Methods for sparse signal recovery using Kalman filtering with embedded pseudo-measurement norms and quasi-norms," *IEEE Trans. Signal Process.*, vol. 58, pp. 2405–2409, 2010.
- [24] D. Zachariah, S. Chatterjee, and M. Jansson, "Dynamic iterative pursuit," *IEEE Trans. Signal Process.*, vol. 60, no. 9, pp. 4967–4972, 2012.
- [25] S. Farahmand, G. B. Giannakis, G. Leus, and Z. Tian, "Tracking target signal strengths on a grid using sparsity," *EURASIP J. Adv. Signal Process.*, vol. 2014, no. 1, p. 7, Jan 2014.
- [26] S. Sundaram and C. N. Hadjicostis, "Delayed observers for linear systems with unknown inputs," *IEEE Trans. Autom. Control*, vol. 52, no. 2, pp. 334–339, 2007.
- [27] J. Tropp, "Just relax: convex programming methods for identifying sparse signals in noise," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 1030–1051, 2006.
- [28] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc., Ser. B*, vol. 58, pp. 267–288, 1996.
- [29] M. J. Wainwright, "Sharp thresholds for high-dimensional and noisy sparsity recovery using ℓ_1 -constrained quadratic programming (Lasso)," *IEEE Trans. Inf. Theory*, vol. 55, no. 5, pp. 2183–2202, 2009.
- [30] R. J. Tibshirani, "The lasso problem and uniqueness," *Electronic Journal of Statistics*, vol. 7, pp. 1456–1490, 2013.
- [31] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, pp. 4655–4666, 2007.
- [32] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2010.
- [33] J. Matamoros, S. M. Fosson, E. Magli, and C. Antón-Haro, "Distributed ADMM for in-network reconstruction of sparse signals with innovations," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 1, no. 4, pp. 225–234, 2015.
- [34] A. Fiandrotti, S. M. Fosson, C. Ravazzi, and E. Magli, "GPU-accelerated algorithms for compressed signals recovery with application to astronomical imagery deblurring," *Int. J. Remote Sens.*, vol. 39, no. 7, pp. 2043–2065, 2018.
- [35] J. J. Fuchs, "Recovery of exact sparse representations in the presence of bounded noise," *IEEE Trans. Inf. Theory*, vol. 51, no. 10, pp. 3601–3608, 2005.